REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
		FINAL/01 FEB 93 TO 31 JAN 95

4. TITLE AND SUBTITLE
OPTIMIZATION PROBLEMS IN MULTITARGET/MULTISENSOR TRACKING

5. FUNDING NUMBERS
61102F 2304/DS

6. AUTHOR(S)
AUBREY B. POORE

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
DEPARTMENT OF MATHEMATICS
COLORADO STATE UNIVERSITY
FORT COLLINS, CO 80523

8. PERFORMING ORGANIZATION REPORT NUMBER
AFOSR-TR- 95 - 0300

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AFOSR/NM
110 DUNCAN AVE, SUTE B115
BOLLING AFB DC 20332-0001

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
F49620-93-1-0133

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION IS UNLIMITED

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

A multi-sensor multi-target tracker based on the use of near optimal and real-time algorithms for data association has been developed.

14. SUBJECT TERMS

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR(SAME AS REPORT)

FINAL TECHNICAL REPORT

for

OPTIMIZATION PROBLEMS IN MULTITARGET/MULTISENSOR TRACKING

AFOSR Grant Number F49620-93-1-0133

by

Aubrey B. Poore
Department of Mathematics
Colorado State University
Fort Collins, CO 80523
Email: poore@math.colostate.edu
Phone: 970-491-6695

Abstract

The ever-increasing demand in surveillance is to produce highly accurate target and track identification and estimation in real-time, even for dense target scenarios and in regions of high track contention. The use of multiple sensors, through more varied information, has the potential to greatly enhance target identification and state estimation. For multitarget tracking, the processing of multiple scans all at once yields the desired track identification and accurate state estimation; however, one must solve an NP-hard data association problem of partitioning observations into tracks and false alarms in real-time. This report summarizes the development of a multisensor-multitarget tracker based on the use of near-optimal and real-time algorithms for the data association problem and is divided into several parts. The first part addresses the formulation of multisensor and multiscan processing of the data association problem as a combinatorial optimization problem. The new algorithms under development for this NP-hard problem are based on a recursive Lagrangian relaxation scheme, construct near-optimal solutions in real-time, and use a variety of techniques such as two dimensional assignment algorithms, a bundle trust region method for the nonsmooth optimization, graph theoretic algorithms for problem decomposition, and a branch and bound technique for small solution components. A brief computational complexity analysis as well as a comparison with some additional heuristic and optimal algorithms is included to demonstrate the efficiency of the algorithms. A two radar system with data supplied by Rome Labs is used to demonstrate the efficiency and robustness of current multisensor-multitarget tracker that is based on these fast data association algorithms.

TABLE OF CONTENTS

1. Introduction. The ever-increasing demand in surveillance is to produce highly accurate target and track identification and estimation in real-time, even for dense target scenarios and in regions of high track contention. The use of multiple sensors, through more varied information, has the *potential* to greatly improve state estimation and track identification. This approach is part of a much broader problem called data fusion, which for military applications is defined as "a multilevel, multifaceted process dealing with the detection, association, correlation, estimation and combination of data and information from multiple sources to achieve refined state and identify estimation, and complete and timely assessments of situation and threat" [64]. The various problems are generally partitioned into three or more levels: (1) fused position (state) and identity, (2) hostile or friendly military situation assessments, and (3) hostile force threat assessments. (Comprehensive discussions can be found in the books of Waltz and Llinas [64] and Hall [26].) Level 1 deals with single and multisource information involving tracking, correlation, alignment, and association by sampling the external environment with multiple sensors and exploiting other available sources. Numerical processes thus dominate Level 1; symbolic reasoning involving various techniques from artificial intelligence permeate Levels 2 and 3. *This report focuses on Level 1 data fusion with the goal being to use multiple sensors to achieve superior state estimation and track identification.*

Sensor fusion systems vary greatly depending on the particular needs of a surveillance system. Key issues in the design of such a system include sensor type (active or passive), sensor location (distributed or collocated), and the level of association, which ranges from sensor level fusion to centralized fusion with hybrids in between. Although there are many such issues, the *central problem in any surveillance system is the data association problem* of partitioning measurements into tracks and false alarms. To explain this data association problem, we must first address the levels of association.

At one extreme is sensor level tracking, wherein each sensor forms tracks from its own measurements and then the tracks from the sensors are fused in a central location. Once the correlation is complete, one then combines the tracks with appropriate modification in the statistics [13]. Compared with central level fusion, the advantages include the reduced communication costs between the sensors and central processing unit and easier data association. The disadvantages are that combined track estimates tend to be worse than in central-level fusion and the error independence assumptions in data association are no longer valid, thereby introducing additional complexity into the problem [11,13]. At the other extreme is *centralized fusion* in which sensors send measurements to a central processing unit where they can be combined to give *superior state estimation* [11] (compared to fusion of sensor level tracks). The difficulties are generally claimed to be data association (*our strength*), communication costs between the sensor and central processing unit, and the loss of the tracking capability if the central processing unit becomes inoperative. In reality, current and proposed sensor fusion systems for any surveillance system make use of both systems. Certainly, one can treat a hybrid of these two systems by sending the observations associated with a track obtained at the sensor level to a central processing unit and treat the association as in centralized fusion [11]. Having explained the level of data association, we now return to a brief overview of the methods of data association for central and some hybrid central-sensor level tracking.

The existing algorithms range from single scan or sequential processing to multiscan processing. Methods for the former include nearest neighbor, one-to-one or few-to-one assignments, and all-to-one assignments as in the joint probabilistic data association (JPDA) [6] in single sensor tracking. Problems involving one-to-one or few-to-one assignments are generally formulated as (two dimensional) assignment or multi-assignment problems for which there are some excellent algorithms [7, 8, 9]. This methodology is real-time but can result in a large number of partial and incorrect assignments, particularly in dense or high contention scenarios,

and thus incorrect track identification. The difficulty is that decisions, once made, are irrevocable, so that there is no mechanism to correct misassociations. The use of all observations in a scan (e.g., JPDA) to update a track moderates the misassociation problem and has been successful for tracking a few targets in dense clutter [6].

Deferred logic techniques consider several data sets or scans of data from multiple sensors all at once in making data association decisions. At one extreme is batch processing in which all observations (from all time) are processed together, but this is computationally too intensive for real-time applications. The other extreme is sequential processing. Deferred logic methods between these two extremes are of primary interest in this work. *The key advantage of this approach is the ability to change data association decisions over several of the most recent scans of data. It is this feature that leads to superior track estimation.* The principal deferred logic method used to track large numbers of targets in low to moderate clutter is called multiple hypothesis tracking (MHT) in which one builds a tree of possibilities, assigns a likelihood score based on Bayesian estimation, develops an intricate pruning logic, and then solves the data association problem by *explicit enumeration schemes.* The fundamental limitation of MHT, as it now exists, is that it is an NP-hard combinatorial optimization problem, so that in dense scenarios and high track contention or with multiple sensor input, the time required to solve this problem optimally can grow exponentially with the size of the problem. This failure is not graceful, i.e., the method is not robust with respect to real-time needs. *Thus to make MHT viable, near-optimal algorithms are needed to solve the data association problems to the noise level in real-time. This is precisely the subject of this research program and report.* The centralized fusion approach is also highly parallelizable and is ripe for the use of parallel computers in the future.

The first topic (Section 2) in this report is the formulation of multisensor and multiscan processing of the data association problem as an NP-hard combinatorial optimization problem. Next, an overview of some of the near-optimal and real-time algorithms for solving this problem is presented in Section 3. The algorithms under development are based on a recursive Lagrangian relaxation scheme, construct near-optimal solutions in real-time, and use a variety of techniques ranging from two dimensional assignment algorithms, a bundle trust region method for the nonsmooth optimization, graph theoretic properties for problem decomposition, and a branch and bound technique for small solution components. These topics are presented in Sections 3 and 4. A brief computational complexity analysis as well as a comparison with some additional heuristic and optimal algorithms is included in Section 5 to demonstrate the efficiency of the algorithms. The existing and on-going software work is discussed in Section 6. Finally, in Section 7, a two radar system with data supplied by Rome Labs is used to demonstrate the efficiency and robustness of current multisensor-multitarget tracker that is based on these fast data association algorithms.

2. Formulation of the Data Association Problem. The goal of this section is to explain the formulation of the data association problem that governs large classes of data association problems in centralized or hybrid centralized-sensor level multisensor/multitarget tracking. The presentation is brief; technical details are presented for both track initiation and maintenance in [49] for nonmaneuvering targets and [52] for maneuvering targets. These works also contain expressions for the likelihood ratios $L_{i_1 \cdots i_N}$ used in the score in the following equations (2.3) and (2.4). The formulation presented here is of sufficient generality to cover the MHT work of Reid [61], Blackman and Stein [10], and modifications by Kurien [34] to include maneuvering targets. As suggested by Blackman [10] this formulation can also be modified to include target features (e.g., size and type) into the scoring function.

The *data association problems for multisensor and multitarget tracking* considered in this work are

generally posed [6,10,35,49,52] as that of *maximizing the posterior probability of the surveillance region* (given the data) according to

$$\text{Maximize} \left\{ \frac{P(\Gamma = \gamma \mid Z^N)}{P(\Gamma = \gamma^0 \mid Z^N)} \;\middle|\; \gamma \in \Gamma^* \right\} \tag{2.1}$$

where Z^N represents N data sets, γ is a partition of indices of the data (and thus induces a partition of the data), Γ^* is the finite collection of all such partitions, Γ is a discrete random element defined on Γ^*, γ^0 is a reference partition, and $P(\Gamma = \gamma \mid Z^N)$ is the posterior probability of a partition γ being true given the data Z^N. The term partition is defined below; however, this framework is currently sufficiently general to cover set packings and coverings [35].

Consider N data sets $Z(k)$ $(k = 1, \ldots, N)$ each of M_k reports $\{z_{i_k}^k\}_{i_k=1}^{M_k}$, and let Z^N denote the cumulative data set defined by

$$Z(k) = \{z_{i_k}^k\}_{i_k=1}^{M_k} \quad \text{and} \quad Z^N = \{Z(1), \ldots, Z(N)\}, \tag{2.2}$$

respectively. In multisensor data fusion and multitarget tracking the data sets $Z(k)$ may represent different classes of objects, and each data set can arise from different sensors. For track initiation the objects are measurements that must be partitioned into tracks and false alarms. In our formulation of track maintenance [49,52], which uses a moving window over time, one data set will be tracks and remaining data sets will be measurements which are assigned to existing tracks, as false measurements, or are assigned to initiating tracks. In sensor level tracking, the objects to be fused are tracks [10]. In centralized fusion [10], the objects may all be measurements that represent targets or false reports, and the problem is to determine which measurements emanate from a common source.

We specialize the problem to the case of set partitioning [49] defined in the following way. First, for notational convenience in representing tracks, we add a *dummy report* z_0^k to each of the data sets $Z(k)$ in (2.2), and define a "track of data" as $\left(z_{i_1}^1, \ldots, z_{i_N}^N\right)$ where i_k and $z_{i_k}^k$ can now assume the values of 0 and z_0^k, respectively. A partition of the data will refer to a collection of tracks of data wherein each report occurs exactly once in one of the tracks of data and such that all data is used up; the occurrence of a dummy report is unrestricted. The dummy report z_0^k serves several purposes in the representation of missing data, false reports, initiating tracks, and terminating tracks [49]. The reference partition is that in which all reports are declared to be false.

Next under appropriate independence assumptions one can show [49]

$$\frac{P(\Gamma = \gamma \mid Z^N)}{P(\Gamma = \gamma^0 \mid Z^N)} \equiv L_\gamma \equiv \prod_{(i_1 \cdots i_N) \in \gamma} L_{i_1 \cdots i_N}, \tag{2.3}$$

$L_{i_1 \cdots i_N}$ is a likelihood ratio containing probabilities for detection, maneuvers, and termination as well as probability density functions for measurement errors, track initiation and termination. Then if $c_{i_1 \cdots i_N} = -\ln L_{i_1 \cdots i_N}$,

$$-\ln \left[\frac{P(\gamma | Z^N)}{P(\gamma^0 | Z^N)} \right] = \sum_{(i_1, \ldots, i_N) \in \gamma} c_{i_1 \cdots i_N}. \tag{2.4}$$

Using (2.3) and the zero-one variable $z_{i_1 \cdots i_N} = 1$ if $(i_1, \ldots, i_N) \in \gamma$ and 0 otherwise, one can then write the

problem (2.1) as the following N-dimensional assignment problem:

$$
\begin{aligned}
\text{Minimize} \quad & \sum_{i_1=0}^{M_1} \cdots \sum_{i_N=0}^{M_N} c_{i_1 \cdots i_N} z_{i_1 \cdots i_N} \\
\text{Subject To:} \quad & \sum_{i_2=0}^{M_2} \cdots \sum_{i_N=0}^{M_N} z_{i_1 \cdots i_N} = 1, \quad i_1 = 1, \ldots, M_1, \\
& \sum_{i_1=0}^{M_1} \cdots \sum_{i_{k-1}=0}^{M_{k-1}} \sum_{i_{k+1}=0}^{M_{k+1}} \cdots \sum_{i_N=0}^{M_N} z_{i_1 \cdots i_N} = 1, \qquad (2.5) \\
& \qquad \text{for} \quad i_k = 1, \ldots, M_k \text{ and } k = 2, \ldots, N-1, \\
& \sum_{i_1=0}^{M_1} \cdots \sum_{i_{N-1}=0}^{M_{N-1}} z_{i_1 \cdots i_N} = 1, \quad i_N = 1, \ldots, M_N \\
& z_{i_1 \cdots i_N} \in \{0, 1\} \text{ for all } i_1, \ldots, i_N,
\end{aligned}
$$

where $c_{0\cdots0}$ is arbitrarily defined to be zero. Here, each group of sums in the constraints represents the fact that each non-dummy report occurs exactly once in a "track of data". One can modify this formulation to include *multiassignments* of one, some, or all the actual reports. The assignment problem (2.5) is changed accordingly. For example, if $z^k_{i_k}$ is to be assigned no more than, exactly, or no less than $n^k_{i_k}$ times, then the " $= 1$" in the constraint (2.5) is changed to " $\leq, =, \geq n^k_{i_k}$," respectively. Modifications for group tracking and multiresolution features of the surveillance region will be addressed in future work. In making these changes, one must pay careful attention to the independence assumptions that need not be valid in many applications.

For *track maintenance*, we use a *sliding window* of N data sets and one data set containing established tracks [49,52]. The formulation is the same as above except that the dimension of the assignment problem is now $N + 1$.

3. Overview of the Lagrangian relaxation algorithms. Having formulated an N-dimensional assignment problem (2.5), we now turn to a description of the Lagrangian relaxation algorithms. The algorithms presented here are the subject of numerous publications [42, 45, 46, 47, 50, 55, 59] and two patents [53,58] and were developed under the current and previous AFOSR grants. The relaxation procedures presented here are based on relaxing an arbitrary number of constraints. Thus, subsection 3.1 presents many of the relaxation properties associated with the relaxation of an n-dimensional assignment problem to an m-dimensional one via a Lagrangian relaxation of $n-m$ sets of constraints. Although any $n-m$ sets can be relaxed the description here is based on relaxing the last $n-m$ sets of constraints and keeping the first m sets. Given either an optimal or suboptimal solution of the relaxed problem, a technique for restoring feasibility to the original n-dimensional problem is presented in subsection 3.2 and an overview of the Lagrangian relaxation algorithm, in subsection 3.3. A presentation of these various algorithms is the subject of forthcoming publications [59, 59, 55] and the thesis of Alexander J. Robertson [60].

The following notation will be used throughout the remainder of the work. Let N be an integer such

that $N \geq 3$ and let $n \in \{3, \ldots, N\}$. The n-dimensional assignment problem is

$$
\begin{aligned}
\text{Minimize} \quad & V_n(z) = \sum_{i_1=0}^{M_1} \cdots \sum_{i_n=0}^{M_n} c^n_{i_1 \cdots i_n} z^n_{i_1 \cdots i_n} \\
\text{Subject To:} \quad & \sum_{i_2=0}^{M_2} \cdots \sum_{i_n=0}^{M_n} z^n_{i_1 \cdots i_n} = 1, \quad i_1 = 1, \ldots, M_1, \\
& \sum_{i_1=0}^{M_1} \cdots \sum_{i_{k-1}=0}^{M_{k-1}} \sum_{i_{k+1}=0}^{M_{k+1}} \cdots \sum_{i_n=0}^{M_n} z^n_{i_1 \cdots i_n} = 1, \\
& \qquad \text{for} \quad i_k = 1, \ldots, M_k \text{ and } k = 2, \ldots, n-1, \\
& \sum_{i_1=0}^{M_1} \cdots \sum_{i_{n-1}=0}^{M_{n-1}} z^n_{i_1 \cdots i_n} = 1, \quad i_n = 1, \ldots, M_n \\
& z^n_{i_1 \cdots i_n} \in \{0, 1\} \text{ for all } i_1, \ldots, i_n.
\end{aligned}
\tag{3.1}
$$

To ensure that a feasible solution of (3.1) always exists, all variables with exactly one nonzero index (i.e., variables of the form $z^n_{0 \cdots 0 i_k 0 \cdots 0}$ for $i_k \neq 0$) are assumed free to be assigned and the corresponding cost coefficients are well-defined.

3.1 The Lagrangian Relaxed Assignment Problem. The n-dimensional assignment problem (3.1) has n sets of constraints. A (M_k+1)-dimensional multiplier vector associated with the k-th constraint set will be denoted by $u^k = (u^k_0, u^k_1, \ldots, u^k_{M_k})^T$ with $u^k_0 = 0$ and $k = 1, \ldots, n$. The n-dimensional assignment problem (3.1) is relaxed to an m-dimensional assignment problem by incorporating $n-m$ of the n sets of constraints into the objective function. Although any constraint set can be relaxed, the description of the relaxation procedure for (3.1) will be based on the relaxation of the last $n-m$ sets of constraints. The *relaxed problem* is

$$
\begin{aligned}
& \Phi_{mn}(u^{m+1}, \ldots, u^n) \equiv \\
& \quad \text{Minimize} \quad \phi_{mn}(z^n; u^{m+1}, \ldots, u^n) \equiv \sum_{i_1=0}^{M_1} \cdots \sum_{i_n=0}^{M_n} c^n_{i_1 \cdots i_n} z^n_{i_1 \cdots i_n} \\
& \qquad\qquad + \sum_{k=m+1}^{n} \sum_{i_k=0}^{M_k} u^k_{i_k} \left[\sum_{i_1=0}^{M_1} \cdots \sum_{i_{k-1}=0}^{M_{k-1}} \sum_{i_{k+1}=0}^{M_{k+1}} \cdots \sum_{i_n=0}^{M_n} z^n_{i_1 \cdots i_n} - 1 \right] \\
& \qquad\qquad \equiv \sum_{i_1=0}^{M_1} \cdots \sum_{i_n=0}^{M_n} \left[c^n_{i_1 \cdots i_n} + \sum_{k=m+1}^{n} u^k_{i_k} \right] z^n_{i_1 \cdots i_n} - \sum_{k=m+1}^{n} \sum_{i_k=0}^{M_k} u^k_{i_k} \\
& \quad \text{Subj. To} \quad \sum_{i_2=0}^{M_2} \cdots \sum_{i_n=0}^{M_n} z^n_{i_1 \cdots i_n} = 1, \quad i_1 = 1, \ldots, M_1, \\
& \qquad \sum_{i_1=0}^{M_1} \cdots \sum_{i_{k-1}=0}^{M_{k-1}} \sum_{i_{k+1}=0}^{M_{k+1}} \cdots \sum_{i_n=0}^{M_n} z^n_{i_1 \cdots i_n} = 1, \\
& \qquad\qquad \text{for} \quad i_k = 1, \ldots, M_k \text{ and } k = 2, \ldots, m, \\
& \qquad z^n_{i_1 \cdots i_n} \in \{0, 1\} \text{ for all } i_1, \ldots, i_n.
\end{aligned}
\tag{3.2}
$$

An optimal (or suboptimal) solution of (3.2) can be constructed from that of an m-dimensional assignment problem. To show this, define for each $(i_1, \ldots, i_m)$ an index $(j_{m+1}, \ldots, j_n) =$

$(j_{m+1}(i_1,\ldots,i_m),\ldots,j_n(i_1,\ldots,i_m))$ and a new cost function $c^m_{i_1\cdots i_m}$ by

$$
\begin{aligned}
(j_{m+1},\ldots,j_n) &= \arg\min\left\{ c^n_{i_1\cdots i_n} + \sum_{k=m+1}^{n} u^k_{i_k} \;\middle|\; i_k = 0,\ldots,M_k \text{ and } k = m+1,\ldots,n \right\} \\
c^m_{i_1\cdots i_m} &= c^n_{i_1\cdots i_m j_{m+1}\cdots j_n} + \sum_{k=m+1}^{n} u^k_{j_k} \text{ for } (i_1,\ldots,i_m) \neq (0,\ldots,0) \qquad (3.3) \\
c^m_{0\cdots 0} &= \sum_{i_{m+1}=0}^{M_{m+1}} \cdots \sum_{i_n=0}^{M_n} \text{Minimum}\left\{ 0, c^n_{0\cdots 0 i_{m+1}\cdots i_n} + \sum_{k=m+1}^{n} u^k_{i_k} \right\}
\end{aligned}
$$

(If $(j_{m+1},\ldots,j_n)$ is not unique, choose the smallest such j_{m+1}, amongst those $(n-m)$-tuples with the same j_{m+1} choose the smallest j_{m+2}, etc., so that $(j_{m+1},\ldots,j_n)$ is uniquely defined.) Using the cost coefficients defined in this way, the following m-dimensional assignment problem is obtained:

$$
\begin{aligned}
\hat{\Phi}_{mn}(u^{m+1},\ldots,u^n) = \text{Minimize} \quad & \hat{\phi}_{mn}(z^m; u^{m+1},\ldots,u^n) \equiv v_m(z^m) \equiv \sum_{i_1=0}^{M_1} \cdots \sum_{i_m=0}^{M_m} c^m_{i_1\cdots i_m} z^m_{i_1\cdots i_m} \\
\text{Subject To:} \quad & \sum_{i_2=0}^{M_2} \cdots \sum_{i_m=0}^{M_m} z^m_{i_1\cdots i_m} = 1, \quad i_1 = 1,\ldots,M_1, \\
& \sum_{i_1=0}^{M_1} \cdots \sum_{i_{k-1}=0}^{M_{k-1}} \sum_{i_{k+1}=0}^{M_{k+1}} \cdots \sum_{i_m=0}^{M_m} z^m_{i_1\cdots i_m} = 1, \qquad (3.4) \\
& \qquad \text{for} \quad i_k = 1,\ldots,M_k \text{ and } k = 2,\ldots,m-1, \\
& \sum_{i_1=0}^{M_1} \cdots \sum_{i_{m-1}=0}^{M_{m-1}} z^m_{i_1\cdots i_m} = 1, \quad i_m = 1,\ldots,M_m \\
& z^m_{i_1\cdots i_m} \in \{0,1\} \text{ for all } i_1,\ldots,i_m.
\end{aligned}
$$

As an aside, observe that any feasible solution z^n of (3.1) yields a feasible solution z^m of (3.4) via the construction

$$
z^m_{i_1\cdots i_m} = \begin{cases} 1 & \text{if } z^n_{i_1\cdots i_n} = 1 \text{ for some } (i_{m+1},\ldots,i_n) \\ 0 & \text{otherwise.} \end{cases}
$$

Thus the m-dimensional assignment problem (3.4) has at least as many feasible solutions of the constraints as the original problem (3.1).

The following theorem states that an optimal solution of (3.2) can be computed from that of (3.4). The converse is contained in Theorem 3.2. Furthermore, if the solution of either of these two problems is ϵ-optimal, then so is the other.

Theorem 3.1. *Let w^m be a feasible solution to problem (3.4) and define w^n by*

$$
\begin{aligned}
w^n_{i_1\cdots i_n} &= w^m_{i_1\cdots i_m} && \text{if } (i_{m+1},\ldots i_n) = (j_{m+1},\ldots,j_n) \text{ and } (i_1,\ldots,i_m) \neq (0,\ldots,0) \\
w^n_{i_1\cdots i_n} &= 0 && \text{if } (i_{m+1},\ldots i_n) \neq (j_{m+1},\ldots,j_n) \text{ and } (i_1,\ldots,i_m) \neq (0,\ldots,0) \\
w^n_{0\cdots 0 i_{m+1}\cdots i_n} &= 1 && \text{if } c^n_{0\cdots 0 i_{m+1}\cdots i_n} + \sum_{k=m+1}^{n} u^k_{i_k} \leq 0 \qquad (3.5) \\
w^n_{0\cdots 0 i_{m+1}\cdots i_n} &= 0 && \text{if } c^n_{0\cdots 0 i_{m+1}\cdots i_n} + \sum_{k=m+1}^{n} u^k_{i_k} > 0
\end{aligned}
$$

Then w^n is a feasible solution of the Lagrangian relaxed problem (3.2) and $\phi_{mn}(w^n; u^{m+1}, \ldots, u^n) = \hat{\phi}_{mn}(w^m; u^{m+1}, \ldots, u^n) - \sum_{k=m+1}^{n} \sum_{i_k=0}^{M_k} u_{i_k}^k$. If, in addition, w^m is optimal for (3.4), then w^n is an optimal solution of (3.2) and $\Phi_{mn}(u^{m+1}, \ldots, u^n) = \hat{\Phi}_{mn}(u^{m+1}, \ldots, u^n) - \sum_{k=m+1}^{n} \sum_{i_k=0}^{M_k} u_{i_k}^k$.

With the exception of one equality being converted to an inequality, the following theorem is a converse of this theorem.

Theorem 3.2. *Let w^n be a feasible solution to problem (3.2) and define w^m by*

$$w_{i_1 \cdots i_m}^m = \sum_{i_{m+1}=0}^{M_{m+1}} \cdots \sum_{i_n=0}^{M_n} w_{i_1 \cdots i_n}^n \text{ for } (i_1, \ldots, i_m) \neq (0, \ldots, 0)$$

$$w_{0 \cdots 0}^m = 0 \text{ if } (i_1, \ldots, i_m) = (0, \ldots, 0) \text{ and } c_{0 \cdots 0 i_{m+1} \cdots i_n}^n + \sum_{k=m+1}^{n} u_{i_k}^k > 0 \text{ for all } (i_{m+1}, \ldots, i_n) \tag{3.6}$$

$$w_{0 \cdots 0}^m = 1 \text{ if } (i_1, \ldots, i_m) = (0, \ldots, 0) \text{ and } c_{0 \cdots 0 i_{m+1} \cdots i_n}^n + \sum_{k=m+1}^{n} u_{i_k}^k \leq 0 \text{ for some } (i_{m+1}, \ldots, i_n).$$

Then w^m is a feasible solution of the problem (3.4) and $\phi_{mn}(w^n; u^{m+1}, \ldots, u^n) \geq \hat{\phi}_{mn}(w^m; u^{m+1}, \ldots, u^n) - \sum_{k=m+1}^{n} \sum_{i_k=0}^{M_k} u_{i_k}^k$. If, in addition, w^n is optimal for (3.2), then w^m is an optimal solution of (3.4), $\phi_{mn}(w^n; u^{m+1}, \ldots, u^n) = \hat{\phi}_{mn}(w^m; u^{m+1}, \ldots, u^n) - \sum_{k=m+1}^{n} \sum_{i_k=0}^{M_k} u_{i_k}^k$, and $\Phi_{mn}(u^{m+1}, \ldots, u^n) = \hat{\Phi}_{mn}(u^{m+1}, \ldots, u^n) - \sum_{k=m+1}^{n} \sum_{i_k=0}^{M_k} u_{i_k}^k$.

3.2 The recovery procedure. The next objective is to explain a *recovery procedure*, i.e., given a feasible (optimal or suboptimal) solution w^m of (3.4) (or w^n of (3.2) constructed via Theorem 3.1), generate a feasible solution z^n of (3.1) which is close to w^m in a sense to be specified. *We first assume that no variables in (3.1) are preassigned to zero; this assumption will be removed shortly.* The difficulty with the solution w^n is that it need not satisfy the last $n-m$ sets of constraints in (3.1). (Note however that if w^m is an optimal solution for (3.4) and w^n (constructed as in Theorem 3.1) satisfies the relaxed constraints, then w^n is optimal for (3.1).) The recovery procedure described here is designed to preserve the 0-1 character of the solution w^m of (3.4) as far as possible: If $w_{i_1 \cdots i_m}^m = 1$ and $i_l \neq 0$ for at least one $l = 1, \ldots, m$, the corresponding feasible solution z^n of (3.1) is constructed so that $z_{i_1 \cdots i_n}^n = 1$ for some $(i_{m+1}, \ldots, i_n)$. By this reasoning, variables of the form $z_{0 \cdots 0 i_{m+1} \cdots i_n}^n$ can be assigned a value of one in the recovery problem only if $w_{0 \cdots 0}^m = 1$. However, variables $z_{0 \cdots 0 i_{m+1} \cdots i_n}^n$ will be treated differently in the recovery procedure in that they can be assigned 0 or 1 independent of the value $w_{0 \cdots 0}^m$. This increases the feasible set of the recovery problem, leading to a potentially better solution.

Let $\{(i_1^j, \ldots, i_m^j)\}_{j=1}^{M_0}$ be an enumeration of indices of w^m (or the first m indices of w^n constructed in Theorem 3.1) such that $w_{i_1^j \cdots i_m^j}^m = 1$ and $(i_1^j, \ldots, i_m^j) \neq (0, \ldots, 0)$. Set $(i_1^0, \ldots, i_m^0) = (0, \ldots, 0)$ for $j = 0$ and define

$$c_{j i_{m+1} \cdots i_n}^{n-m+1} = c_{i_1^j \cdots i_m^j i_{m+1} \cdots i_n}^n \quad \text{for } i_k = 0, \ldots, M_k;\ k = m+1, \ldots, n;\ \ j = 0, \ldots, M_0. \tag{3.7}$$

Let Y denote the solution of the $(n-m+1)$-dimensional assignment problem

$$\begin{aligned}
\text{Minimize} \quad & \sum_{j=0}^{M_0} \sum_{i_{m+1}=0}^{M_{m+1}} \cdots \sum_{i_n=0}^{M_n} c^{n-m+1}_{j i_{m+1} \cdots i_n} y_{j i_{m+1} \cdots i_n} \\
\text{Subject To} \quad & \sum_{i_{m+1}=0}^{M_{m+1}} \cdots \sum_{i_n=0}^{M_n} y_{j i_{m+1} \cdots i_n} = 1, \quad j = 1, \ldots, M_0, \\
& \sum_{j=0}^{M_0} \sum_{i_{m+1}=0}^{M_{m+1}} \cdots \sum_{i_{k-1}=0}^{M_{k-1}} \sum_{i_{k+1}=0}^{M_{k+1}} \cdots \sum_{i_n=0}^{M_n} y_{j i_{m+1} \cdots i_n} = 1, \\
& \qquad \text{for} \quad i_k = 1, \ldots, M_k \text{ and } k = m+1, \ldots, n-1, \\
& \sum_{j=0}^{M_0} \sum_{i_{m+1}=0}^{M_{m+1}} \cdots \sum_{i_{n-1}=0}^{M_{n-1}} y_{j i_{m+1} \cdots i_n} = 1, \quad i_n = 1, \ldots, M_n, \\
& y_{j i_{m+1} \cdots i_n} \in \{0,1\} \text{ for all } j, i_{m+1}, \ldots, i_n.
\end{aligned} \tag{3.8}$$

The recovered feasible solution z^n of (3.1) corresponding to the multiplier set $\{u^m, \ldots, u^n\}$ is then defined by

$$z^n_{i_1 \cdots i_n} = \begin{cases} 1 & \text{if } (i_1, \ldots, i_m) = (i^j_1, \ldots, i^j_m) \text{ for some } j = 0, \ldots, M_0 \text{ and } Y_{j i_{m+1} \cdots i_n} = 1 \\ 0 & \text{otherwise.} \end{cases} \tag{3.9}$$

This recovery procedure is valid as long as all cost coefficients c^n are defined and all zero-one variables in z^n are free to be assigned. Modifications are necessary for sparse problems. If the cost coefficient $c^n_{i^j_1 \cdots i^j_m i_{m+1} \cdots i_n}$ is well defined and the zero-one variable $z^n_{i^j_1 \cdots i^j_m i_{m+1} \cdots i_n}$ is free to be assigned to zero or one, then define $c^{n-m+1}_{j i_{m+1} \cdots i_n} = c^n_{i^j_1 \cdots i^j_m i_{m+1} \cdots i_n}$ as in (3.7) with $z^{n-m+1}_{j i_{m+1} \cdots i_n}$ being free to be assigned. Otherwise, $z^{n-m+1}_{j i_{m+1} \cdots i_n}$ is preassigned to zero or the corresponding arc is not allowed in the feasible set of arcs. To ensure that a feasible solution exists, we now only need ensure that the variables $z^{n-m+1}_{j 0 \cdots 0}$ are free for $j = 0, 1, \ldots, M_0$. (Recall that all variables of the form $z^n_{0 \cdots i_k \cdots 0}$ are free (to be assigned) and all coefficients of the form $c^n_{0 \cdots i_k \cdots 0}$ are well defined for $k = 1, \ldots, n$.) This is accomplished as follows: If the cost coefficient $c^n_{i^j_1 i^j_2 \cdots i^j_m 0 \cdots 0}$ is well defined and $z^n_{i^j_1 i^j_2 \cdots i^j_m 0 \cdots 0}$ is free, then define $c^{n-m+1}_{j 0 \cdots 0} = c^n_{i^j_1 i^j_2 \cdots i^j_m 0 \cdots 0}$ with $z^{n-m+1}_{j 0 \cdots 0}$ being free. Otherwise, since all variables of the form $z^n_{0 \cdots i_k \cdots 0}$ are known feasible and have well-defined costs, put $c^{n-m+1}_{j 0 \cdots 0} = \sum_{k=1 i^j_k \neq 0}^{m} c^n_{0 \cdots 0 i^j_k 0 \cdots 0}$.

3.3. Summary of the Lagrangian Relaxation Algorithm. Starting with the N-dimensional assignment problem (3.1), i.e. $n = N$, the algorithm described below is recursive in that the N-dimensional assignment problem is relaxed to an m-dimensional one by incorporating $(n-m)$ sets of constraints into the objective function using a Lagrangian relaxation of this set. This problem is maximized with respect to the Lagrange multipliers, and a good suboptimal solution to the original problem is recovered using an $(n-m+1)$-dimensional assignment problem. Each of these two (the m-dimensional and the $(n-m+1)$-dimensional assignment problems) can be solved in a similar manner. Here we describe one loop in this procedure.

The Lagrangian Relaxation Algorithm for the n-Dimensional Assignment Problem

Assume $N \geq 3$ and choose $n \in \{3, \ldots, N\}$ and $2 \leq m < n$. To obtain a near-optimal solution of the n-dimensional assignment problem (3.1), proceed as follows:

A. Initialization: Choose an initial approximation $\{u^{m+1}_0, \ldots, u^n_0\}$.

B. Use a non-smooth optimization technique (see subections 4.2 and 4.3) to maximize Φ_{mn} in (3.2), i.e.,

$$\text{Maximize } \left\{ \Phi_{mn}(u^{m+1}, \ldots, u^n) \;\middle|\; u^k \in \mathbb{R}^{M_k+1} \;;\; k = m+1, \ldots, n \right\} \tag{3.10}$$

C. Given an optimal solution $(\bar{u}^{m+1}, \ldots, \bar{u}^n)$ of (3.10), compute an optimal solution $\bar{z}^m$ of (3.4).

D. Using the procedure described in Section 3.2, solve the $(n-m+1)$-dimensional assignment problem (3.8) to recover a feasible solution z^n of the n-dimensional assignment problem (3.1).

3.4. Comments on the Various Algorithms.

Of the many algorithms descibed above we dicuss them as three different classes of algorithms. The *first* is from our earlier work [42, 50] and is that in which one set of constraints is relaxed, yielding an $m = n - 1$ dimensional problem by incorporating one set of constraints into the objective function via the Lagrangian. We maximize the relaxed problem with respect to the corresponding Lagrange multipliers and then reconstruct a feasible solution to the n-dimensional problem using a two-dimensional assignment problem. This first relaxation scheme describes the framework for our earlier algorithm work [42, 50] and also incorporates the algorithm presented in [15]. The algorithm of Deb et.al. is a special case of ours in that only one nonsmooth iteration is taken at each inner relaxed m-dimensional level, no merit function is implemented, and no decomposition is performed. Their nonsmooth optimization method is, however, different from those that we use.

The *second* algorithm is somewhat of a mirror image of the first in that $n - 2$ sets of constraints are relaxed, yielding an $m = 2$ dimensional problem. A feasible solution to the n dimensional problem is then recovered using an $n - 1$ dimensional problem. In this case the function values and subgradients of $\Phi_{2n}(u^3, \ldots, u^n)$ can be computed optimally via a two dimensional assignment problem. The significant advantage here is that there is no need for the merit or auxiliary function as discussed in subsection 4.1 and all function values and subgradients used in the nonsmooth maximization process are computed exactly (i.e., optimally). Problem decomposition is now carried out for the n dimensional problem; however, decomposition of the $n - 1$ dimensional recovery problem (and all lower order recovery problems) is performed only after the problem is formulated.

Between these two algorithms are a host of different relaxation schemes based on relaxing $n - m$ sets of constraints to an m-dimensional problem $(2 < m < n)$, but these all have the same difficulties as the first algorithm in that the relaxed problem is an NP-hard problem. To resolve this difficulty, we use an auxiliary or merit function Ψ_{mn} as described in subsection 4.2. For the case $m < n - 1$, the recovery procedure is still based on an NP-hard $(n - m + 1)$-dimensional assignment problem. The decomposition techniques discussed in subsection 4.7 are based on identifying the assignment problem with a layered graph and then finding disjoint components of this graph. In general, all relaxed problems can be decomposed prior to any nonsmooth computations because their structure stays fixed throughout the algorithm. All recovery problems cannot be decomposed until they are formulated, as their structure changes as the solutions to the relaxed problems change.

4. Algorithm Details and Refinements. Many aspects of the algorithm presented in Section 3 require further explanation on how various problems are solved. These along with the many refinements that can significantly increase the speed of the relaxation algorithm are explained in this section.

4.1 Maximization of the Nonsmooth Function $\Phi_{mn}(u^{m+1}, \ldots, u^n)$. One of the key steps in the Lagrangian relaxation algorithm in subsection 3.3 is the solution of the problem

$$\text{Maximize } \{\Phi_{mn}(u^{m+1}, \ldots, u^n) \mid u^k \in \mathbb{R}^{M_k+1} \ ; \ k = m+1, \ldots, n\} \tag{4.1}$$

where $u_0^k = 0$ for all $k = m + 1, \ldots, n$. In this subsection we discuss several aspects of this problem. We first show that this is a problem of nonsmooth optimization and then briefly explain the computation of the

required subgradients. Unfortunately, the evaluation of Φ_{mn} requires the evaluation of an NP-hard problem for $m > 2$. Thus for real time needs, we address this difficulty at the end of the subsection. The actual nonsmooth optimization methods are presented in the next subsection.

That (4.1) is a problem of nonsmooth optimization is the subject of

Theorem 4.1. *Let $u^{m+1}, \ldots, u^n$ be multiplier vectors associated with the $(m+1)^{st}$ through the n^{th} set of constraints in (3.1), let Φ_{mn} be as defined in (3.2), let $V_n(\bar{z}^n)$ be the objective function value of the n-dimensional assignment problem in equation (3.1), let z^n be any feasible solution of (3.1), and let $\bar{z}^n$ be an optimal solution of (3.1). Then, $\Phi_{mn}(u^{m+1}, \ldots, u^n)$ is piecewise affine, concave and continuous in $\{u^{m+1}, \ldots, u^n\}$ and*

$$\Phi_{mn}(u^{m+1}, \ldots, u^n) \leq V_n(\bar{z}^n) \leq V_n(z^n), \tag{4.2a}$$

Furthermore,

$$\Phi_{m-1,n}(u^m, u^{m+1}, \ldots, u^n) \leq \Phi_{mn}(u^{m+1}, \ldots, u^n) \tag{4.2b}$$

for all $u^k \in \mathbb{R}^{M_k+1}$ with $u_0^k = 0$ and $k = m, \ldots, n$.

Thus the problem of maximizing Φ_{mn} is one of nonsmooth optimization.

Most of the algorithms for non-smooth optimization are based on generalized gradients called *subgradients*, given by the following definition.

Definition 4.2. *At $u = (u^{m+1}, \ldots, u^n)$ the set $\delta\Phi_{mn}(u)$ is called a subdifferential of Φ_{mn} and is defined by*

$$\begin{aligned} \delta\Phi_{mn}(u) = \{q \in \mathbb{R}^{M_{m+1}+1} \times \cdots \times \mathbb{R}^{M_n+1} \mid \Phi_{mn}(w) - \Phi_{mn}(u) \leq \\ q^T(w-u) \; \forall \; w \in \mathbb{R}^{M_{m+1}+1} \times \cdots \times \mathbb{R}^{M_n+1}\}. \end{aligned} \tag{4.3}$$

A vector $q \in \delta\Phi_{mn}(u)$ is called a subgradient.

If z^n is an optimal solution of (3.2) computed during evaluation of $\Phi_{mn}(u)$, differentiating Φ_{mn} with respect to $u_{i_n}^n$ yields the following i_n-th component of a subgradient g of $\Phi_{mn}(u)$

$$\begin{aligned} g_0^k &= 0 \\ g_{i_k}^k &= \sum_{i_1=0}^{M_1} \cdots \sum_{i_{k-1}=0}^{M_{k-1}} \sum_{i_{k+1}=0}^{M_{k+1}} \cdots \sum_{i_n=0}^{M_n} z_{i_1 \cdots i_n}^n - 1 \text{ for } i_k = 1, \ldots, M_k \text{ and } k = m+1, \ldots, n. \end{aligned} \tag{4.4}$$

If z^n is the unique optimal solution of (3.2), $\delta\Phi_{mn}(u) = \{g\}$, and Φ_{mn} is differentiable at u. If the optimal solution is not unique, then there are finitely many such solutions, say $z^n(1), \ldots, z^n(K)$. Given the corresponding subgradients, $g^1, \ldots, g^K$, the subdifferential $\delta\Phi(u)$ is the convex hull of $\{g^1, \ldots, g^K\}$ [24].

4.2 The use of a merit or auxiliary function. For real-time needs, one must address the fact that the nonsmooth optimization problem (4.1) requires the solution of an NP-hard problem for $m > 2$. One approach to this problem is to use the following merit or auxiliary function to decide whether a function value has increased or decreased sufficiently in the line search or trust region methods:

$$\Psi_{mn}(\bar{u}^3, \ldots, \bar{u}^m; u^{m+1}, \ldots, u^n) = \begin{cases} \Phi_{mn}(u^{m+1}, \ldots, u^n) & \text{if } m = 2 \\ & \text{or (3.2) is solved optimally,} \\ \Phi_{2n}(\bar{u}^3, \ldots, \bar{u}^m; u^{m+1}, \ldots, u^n) & \text{otherwise.} \end{cases} \tag{4.5}$$

where the multipliers $\bar{u}^3, \ldots, \bar{u}^m$ that appear in lower order relaxations used to construct (suboptimal) solutions of the m-dimensional relaxed problem (3.2) have been explicitly included. Note that Ψ is well-defined since (3.4) can always be solved optimally if $m = 2$. For sufficiently small problems (3.2) or (3.4), one can more efficiently solve the NP-hard problem by branch and bound. This is the reason for the inclusion of the first case; otherwise, the relaxed function Φ_{2n} to guide the nonsmooth optimization phase. That the merit function provides a lower bound for the optimal solution follows directly from Theorem 4.1 and

Theorem 4.3. *Given the definition of* Ψ_{mn} *in (4.5)*

$$\Psi_{mn}(\bar{u}^3, \ldots, \bar{u}^m; u^{m+1}, \ldots, u^n) \leq \Phi_{mn}(u^{m+1}, \ldots, u^n) \text{ for all multipliers } \bar{u}^3, \ldots, \bar{u}^m, u^{m+1}, \ldots, u^n. \quad (4.6)$$

The actual function value used in the optimization phase is Ψ_{mn}; however, the subgradients are computed as in (4.4), but with the solution $z^n_{i_1 \cdots i_n}$ being a suboptimal solution constructed from a relaxation procedure applied to the m-dimensional problem. Again, the use of these lower order relaxed problems is the reason for the inclusion of the multipliers $\bar{u}^3, \ldots, \bar{u}^m$.

To explain how the merit function is used, suppose we have a current multiplier set $(u^{m+1}_{\text{old}}, \ldots, u^n_{\text{old}})$ and we wish to update to a new multiplier set $(u^{m+1}_{\text{new}}, \ldots, u^n_{\text{new}})$ via $(u^{m+1}_{\text{new}}, \ldots, u^n_{\text{new}}) = (u^{m+1}_{\text{old}}, \ldots, u^n_{\text{old}}) + (\Delta u^{m+1}, \ldots, \Delta u^n)$. Then we compute $\Psi_{mn}(\bar{u}^3_{\text{old}}, \ldots, \bar{u}^m_{\text{old}}; u^{m+1}_{\text{old}}, \ldots, u^n_{\text{old}})$ where $(\bar{u}^3_{\text{old}}, \ldots, \bar{u}^m_{\text{old}})$ is obtained during the relaxation process used to compute a high quality solution to the relaxed m-dimensional assignment problem (3.2) at $(u^{m+1}, \ldots, u^n) = (u^{m+1}_{\text{old}}, \ldots, u^n_{\text{old}})$. The decision to accept $(u^{m+1}_{\text{new}}, \ldots, u^n_{\text{new}})$ is then based on $\Psi_{mn}(\bar{u}^3_{\text{old}}, \ldots, \bar{u}^m_{\text{old}}; u^{m+1}_{\text{old}}, \ldots, u^n_{\text{old}}) < \Psi_{mn}(\bar{u}^3_{\text{new}}, \ldots, \bar{u}^m_{\text{new}}; u^{m+1}_{\text{new}}, \ldots, u^n_{\text{new}})$ or some other stopping criteria commonly used in line searches. Again, $(\bar{u}^3_{\text{new}}, \ldots, \bar{u}^m_{\text{new}})$ represents the multiplier set used in the lower level relaxation procedure to construct a high quality feasible solution to the m-dimensional relaxed problem (3.2) at $(u^{m+1}, \ldots, u^n) = (u^{m+1}_{\text{new}}, \ldots, u^n_{\text{new}})$. The point is that each time one changes $(u^{m+1}, \ldots, u^n)$ and uses the merit function $\Psi_{mn}(\bar{u}^3, \ldots, \bar{u}^m; u^{m+1}, \ldots, u^n)$ for comparison purposes, one must generally change the lower level multipliers $(\bar{u}^3, \ldots, \bar{u}^m)$.

An illustration of this merit function for $m = n - 1$ is given in the work of Poore and Rijavec [50].

4.3. Nonsmooth Optimization Methods. By Theorem 4.1 the function $\Phi_{mn}(u)$ is a continuous, piecewise affine, and concave, so that the negative of $\Phi_{mn}(u)$ is convex. Thus the problem of maximizing $\Phi_{mn}(u)$ is one of nonsmooth optimization. There is a large amount of literature on such problems [29, 30, 33, 62, 63, 65]. Suffice it to say that we have tried a variety of methods including subgradient methods [63], bundle methods [29, 30, 33], and the recent bundle trust method of Schramm and Zowe [62]. We have determined that for a fixed number of nonsmooth iterations, say, ten, the bundle-trust method provides good quality solutions with the fewest number of function and subgradient evaluations of all the methods, and is therefore our currently recommended approach.

4.4. The Two Dimensional Assignment Problem. The forward/reverse auction algorithm of Bertsekas, Castañon, and Tsaknakis [9] is used to solve the many relaxed two dimensional problems that occur in the course of execution.

4.5. Initial Multipliers and Hot Starts. The effective use of "hot starts" is fundamental for real-time applications. A good initial set of multipliers can significantly reduce the number of nonsmooth iterations (and hence the number of Φ_{mn} evaluations) required for a high quality recovered solution. A presentation of these techniques can be found in the thesis of Robertson [60].

4.6 Local Search Methods. Given a feasible solution of the multidimensional assignment problem, one can consider local search procedures to improve this result [38]. A discussion of these methods is presented in the thesis of Robertson [60].

4.7. Problem Decomposition. The algorithm described thus far is all based on relaxation. Due to the sparsity of the problems, one can frequently decompose the problem into a collection of disjoint components each of which can be solved independently. Due to the setup costs of Lagrangian relaxation, a branch and bound procedure is generally more efficient for small components, say four or five feasible arcs. Otherwise, we use the relaxation procedures described above.

Perhaps the easiest way to view our decomposition method is to view the reports or measurements as a layered graph. A vertex is associated with each observation point, and an edge is allowed to connect two vertices only if the two observations belong to at least one feasible track of observations. Given this graph, the decomposition problem can then be posed as that of identifying the connected subcomponents of a graph which can be accomplished by constructing a spanning forest via a depth first search algorithm [1].

The orginal relaxation problem is decomposed first. All relaxed assignment problems can be decomposed *a priori* and all recovery problems can be decomposed only after they are formulated. Hence, in the n-to-$(n-1)$ case, we have $n-2$ relaxed problems that can all be decomposed initially, and the recovery problems are not decomposed (since they are all two dimensional). In the n-to-2 case, we have only one relaxed problem that can be decomposed initially. This case yields $n-3$ recovery problems, which can be decomposed only after they are formulated.

4.8 Use of Invariance and Cost Shifting. A heuristic that can add significant speed to the overall relaxation procedure is to *use the following invariance principle to shift the most negative cost to zero.* This work is discussed in the work of Rijavec and Poore [50].

5. Numerical Performance of the Algorithm. The Lagrangian relaxation algorithm in subsection 3.3 with $m = 2$ is sufficiently general to encompass a wide range of dense and sparse multidimensional assignment problems. We should note a few features of our implementation that have an impact on the performance numbers that follow. First, the problems considered do not decompose. Second, we store only free variables (as opposed to a multidimensional matrix). The algorithm incorporates the bundle-trust method [62] to solve the nonsmooth optimization problem (4.1) and the forward/reverse asymmetric auction algorithm [8, 9] to solve the relaxed two-dimensional assignment problem (3.4). The program currently executes the same number of nonsmooth optimization steps for each assignment problem for which $n > 2$. When the solver reaches a 3-dimensional recovery problem, a 2-dimensional recovery procedure is executed for each nonsmooth iteration.

As we now demonstrate for a particular class of problems, the execution time for the algorithm in subsection 3.3 with $m = 2$ is linear in the number of free or feasible variables and in the number of nonsmooth iterations. This is due to the fact that the dominant computational part of the algorithm is the evaluation of the relaxed two-dimensional cost coefficients as specified in (3.3). The problem class that we have chosen is 4-dimensional with $M_k = 25$ on average for $k = 1, \ldots, 4$. The cost coefficients are uniformly random on the interval $[-100, -1]$. The times are averages over 100 randomly generated problems, with the algorithm running on an IBM RISC 6000/550. Our first test was to determine execution time as a function of the number of free variables for a fixed number of nonsmooth iterations. Our second test was to determine execution time as a function of the number of nonsmooth iterations, with a fixed number of free variables. Results are shown in Figures 5.1 and 5.2.

In these graphs, the lines correspond to a linear least squares fit of the data provided. In Figure 5.1, the upper line represents the overall execution time and the lower line represents the time spent in evaluation of the relaxed cost coefficients. Figure 5.1 shows that the growth in execution time is a linear function of the number of free variables, and the evaluation of the relaxed cost coefficients (3.3) clearly dominates overall execution. Also, the fact that the gap grows slightly between the two lines as the number of arcs increases shows that the other algorithm components grow slightly in execution time. Figure 5.2 shows that the growth in execution is a linear function of the number of nonsmooth iterations. We can therefore conclude that the overall performance time of algorithm in subsection 3.3 with $m = 2$ is a linear function of the number of free variables.

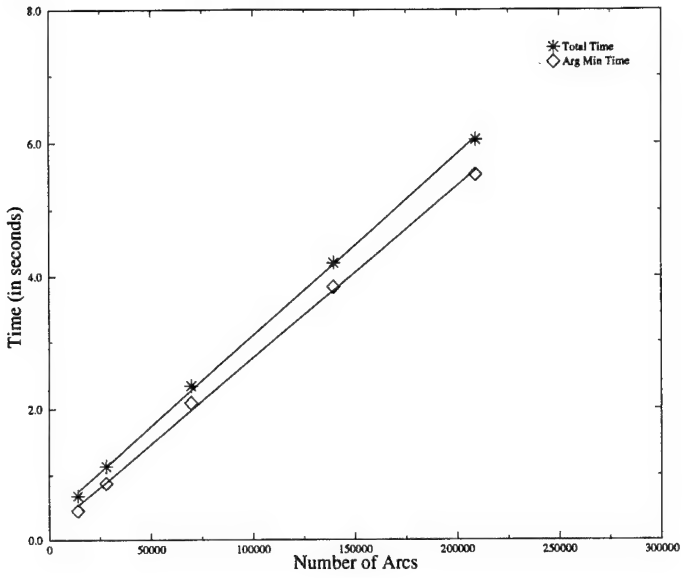

Figure 5.1 – Number of Arcs vs Speed

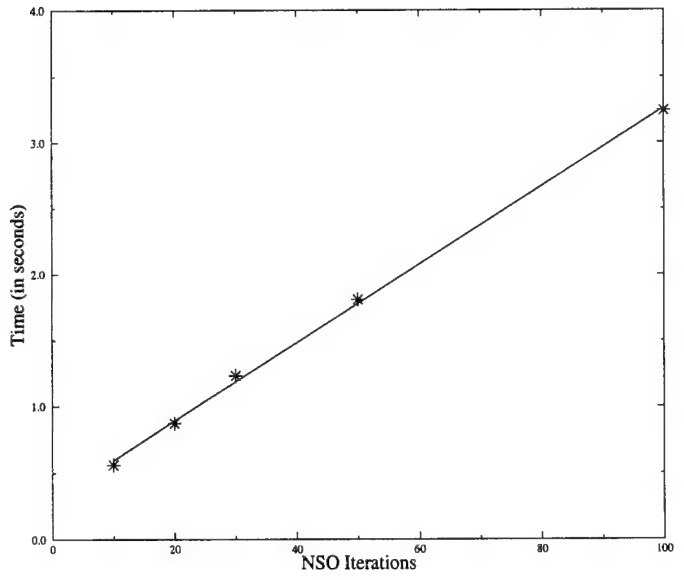

Figure 5.2 – Number of Nonsmooth Iterations vs Speed

Next we consider the comparison of two different suboptimal methods (relaxation and randomized greedy) to the optimal solution provided by branch-and-bound, run for 50 randomly generated problems.

The randomized greedy was the best of serveral other heuristic methods including greedy and max-regret, and thus we display on the results for this randomized greedy. The problems were all four dimensional, with average size $M_k = 7$, and costs uniformly generated on the interval $[1, 100]$. The relaxation method was executed to convergence, which averaged 200 nonsmooth iterations. All objective function values are normalized to the optimal (100) solution. The graph shows the superior quality of the recovered feasible solutions, and also demonstrates the small approximate duality gaps provided by relaxation.

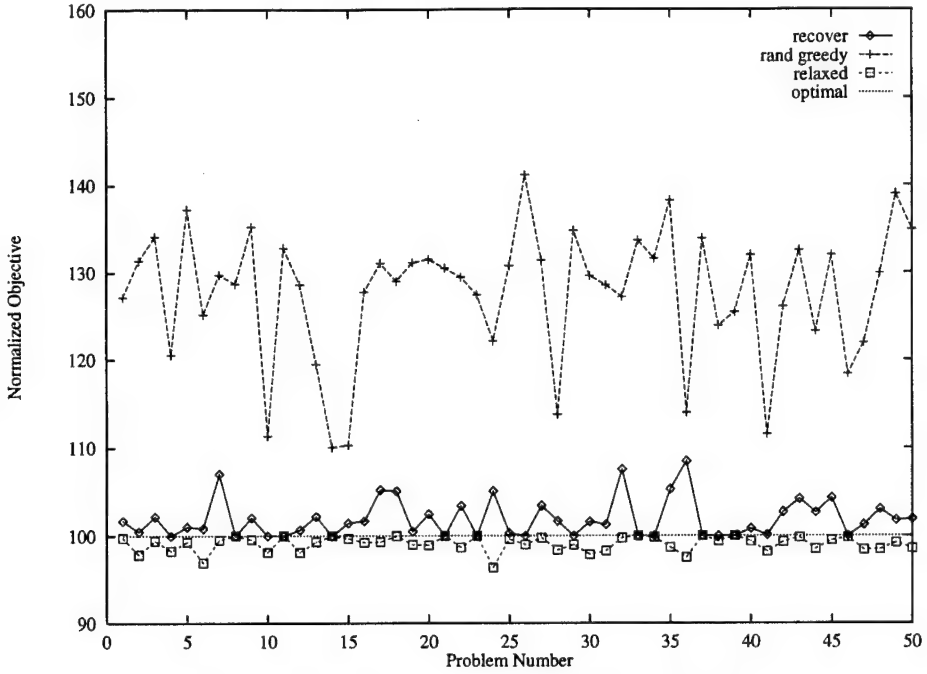

Figure 5.3 – REL vs. RGR, 25% of arcs feasible

6. Tracking Software. Since the beginning of 1994, we have completely redesigned and rewritten our 1993 multitarget tracker to include multisensor processing with new and improved data structures. The current program, which is about twenty thousand lines of C code, is modular and is designed for adaptation to different sensor fusion scenarios: moving vs. stationary platforms, collocated vs. distributed sensors, passive and/or active sensors, and handles asynchronous, out of sequence and delayed measurements/data. New maneuvering target algorithms now utilize the unique capability for fast re-association over multiscan window to switch between multiple models for the target dynamics. This software is, however, not available at the time this report was written.

7. A Two Radar System with Rome Labs Data. This section describes a tracking problem based on six and a half minutes of real data observed by radars at Dansville and Remsen in upstate New York. The Remsen radar is approximately 20 NM Northeast of Rome Labs while the Dansville radar resides about 100 NM West Southwest of Rome Labs. Both radars are L-band radars. They consist of a search radar and a Beacon Interrogator with a common digitizer set. The antenna of the radar and the beacon are collocated and move together. The JSS site characteristics of interest are:

	DANSVILLE, NY	REMSEN, NY
Antenna speed	6 rpm	5 rpm
Frequency	1280-1350 MHz	1250 - 1350 MHz
Prf	350 Hz	340 Hz
Pulse width	1.8 microsec	6 microsec
Vertical beamwidth	3.75°	5.4°
Horizontal beamwidth	1.2°	1.3°
Range	200 NM	200 NM

The data is received by specialized hardware called Data Interface Unit (DIU) where the asynchronous data is formatted into scans from each radar and sent on to files on a MicroVAX II or used as input to fusion algorithms. The detection times are generated by the combining of Universal Time code (UTC) with the delta time supplied by each input detection message. The two types of sensors collocated at the radar site detect two types of targets: radar and beacon. The radar system picks up the reflected signal, or "skin paint," from a non-cooperative target. Note that the signal can also be reflected by weather, atmospheric conditions, terrain, etc., causing false targets or clutter. Various canceling mechanisms can remove much of this clutter, but the remainder, or "clutter residue," appears to the system as radar targets. The Beacon IFF (Identification Friend of Foe) system relies on a transponder onboard the cooperative aircraft to respond to "who are you?" and "how high are you?" queries from the site. The Beacon system is thus of limited utility in a threat scenario, except to identify targets that are probably not a threat. It is also useful for providing "ground truth" for system testing when a cooperative target is used for a simulated threat.

The beacon responds with an IFF response (squawk) and the altitude of the target. This response is a four digit number that is set manually by the target aircraft denoting the identity of the aircraft. This number may change during flight under direction of air traffic control. If either of the last two digits are non zero, the code is said to be discrete and unique to one aircraft in the sky. Usually all aircraft that are flying visual flight rules (VFR) are assigned the same number, 1200. Some aircraft do not have transponders. The altitude of the aircraft is sent with the beacon response. The altitude is a barometric pressure reading. A barometric altitude correction is needed for altitudes below 18,000 feet to correct it to true feet above Mean Sea Level. An apparent discontinuity in the altitude could occur as the target climbs or descends through the "transition altitude" of 18,000 feet. In summary, there are three types of target detections reported:

Search Radar Detection: This is a detection by the radar system only. The target could be oriented such that the beacon signal did not activate the target's beacon or the target's beacon system could be turned off, but the target returned a radar signal (skin return). This detection could also be a false alarm.

Beacon Reinforced Target: This detection report results when both the beacon receives a transponder return and the radar receives a skin return at the same position.

Beacon Only Target: A signal received from the target's beacon which is not correlated with a radar signal. This can occur when the target aircraft's radar cross section is such that the radar skin return is below the detection threshold, but the one way energy of the beacon signal is strong enough to trigger the target aircraft's transponder.

More than sixty aircraft are observed in the period covered by the data. There is also some clutter, both random and stationary, probably due to ground features. Almost all the observations in the problem come from the overlap region(i.e., the region observed by both radars). Several aircraft, however, are shown

outside of that area, especially north of Remsen. Due to the terrain features, and possibly other causes, a number of targets *within* the overlap region are observed by one radar only through all or part of their track life. For some targets, even the coverage by one sensor is intermittent.

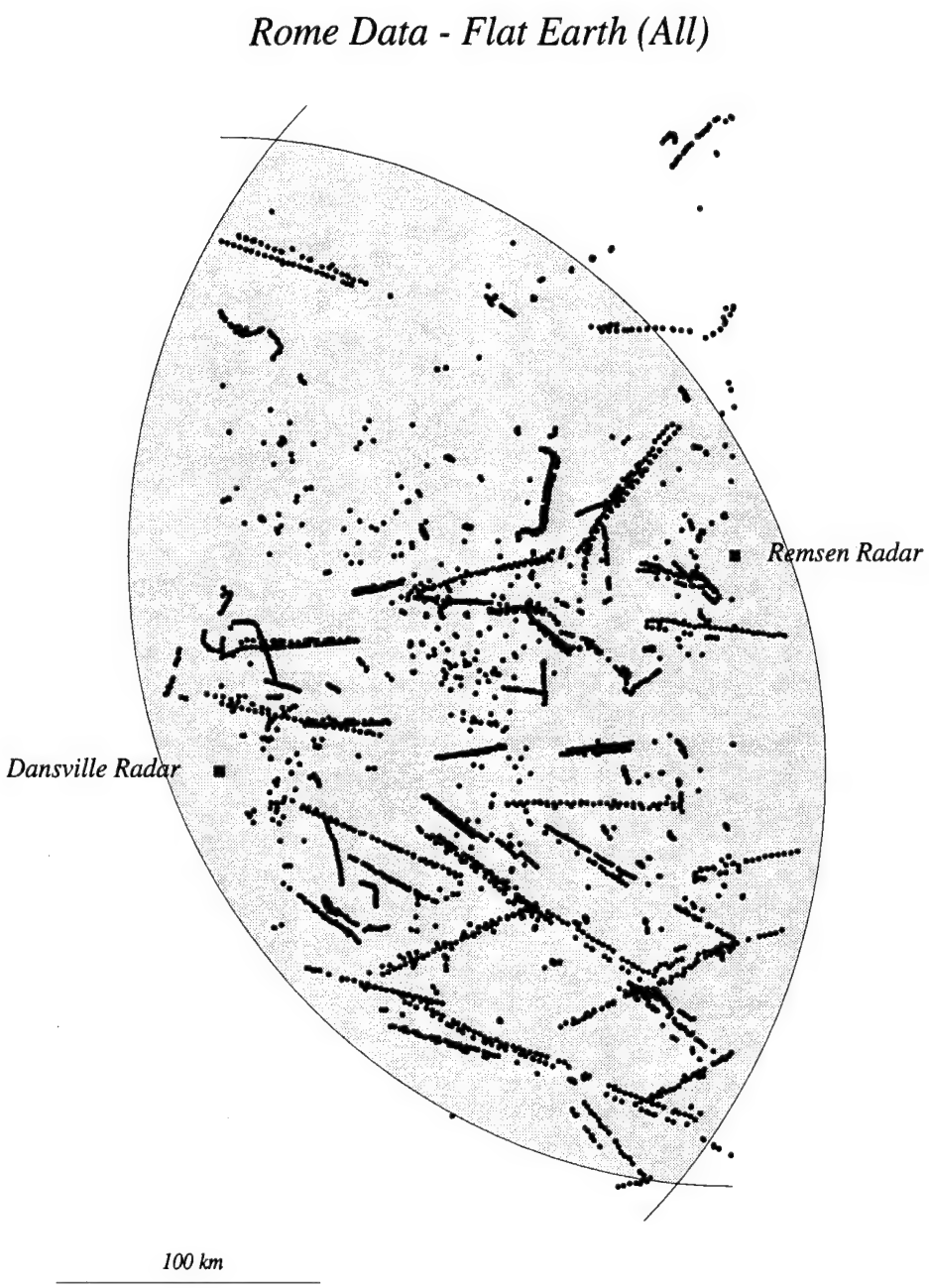

Figure 7.1.

At the ranges present in the problem, the curvature of the Earth becomes an issue and the "flat earth assumption" leads to significant misalignment problems. Figure 7.1 shows the observations in the tracking plane generated by ignoring the Earth's curvature. It can be seen that the observations from the two sensors form parallel lines for each target observed by both sensors.

Figure 7.2 shows that the curvature of the Earth must be taken into account. Mulholland and Stout [36] describe a stereographic projection algorithm used in the National Aerospace System. The tradeoffs between the different projections are described in some detail by Goldenberg and Wolf, [25] and the stereographic projection is determined to be superior to other alternatives. The stereographic projection was thus chosen to transform the tracking problem into a Cartesian coordinate system.

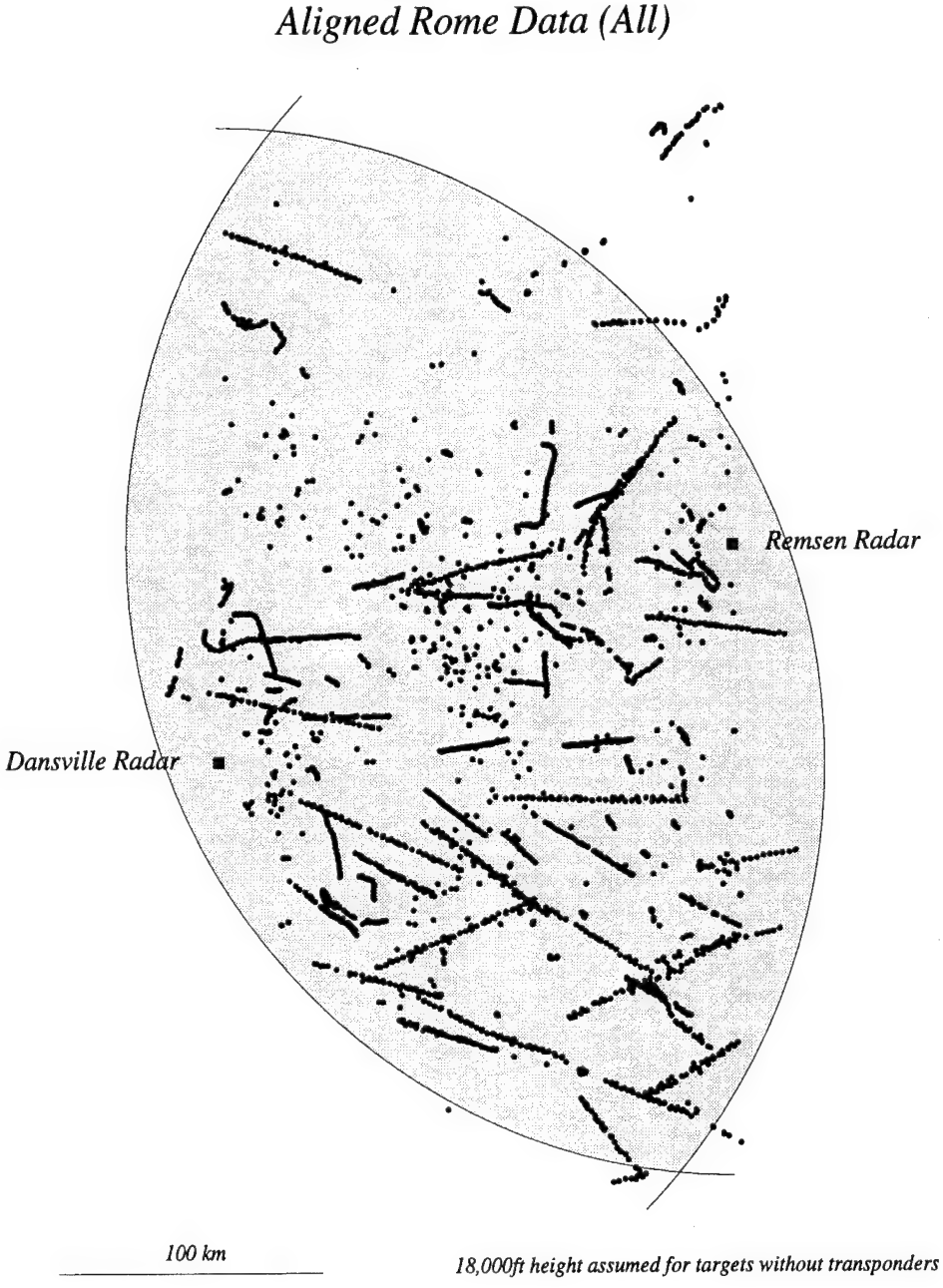

Figure 7.2.

The tracking coordinate system was chosen to be 3D Cartesian coordinate system, with the origin midpoint between the radar sites, and the height as average of the height of the radar sites. The positive x axis pointed east, while the positive y axis pointed north.

Even after applying the stereographic projection, a slight misalignment remained in the data. It was found, however, that applying a 0.5 kilometer correction to the range of the Remsen radar and 0.1° correction to the azimuth of the Dansville radar removed almost all the misalignments. The aligned observations are shown in Figure 7.3. When more data becomes available, a more thorough analysis of the alignment problems can be made using system identification algorithms.

Computing a stereographic projection of an observation with known height is straightforward. If, however, only range and azimuth are available, the value of the height parameter must be assumed. This was taken to be 18,000 feet. The additional inaccuracy in the data resulting from the height assumption must be addressed in the tracking algorithms.

7.2 Numerical Solution. This section will give a brief discussion of the algorithms used and experience gained in solving the two radar tracking problem described in the preceding section. In a general multisensor tracking problem, the target space would likely be partitioned according to the sensor coverage, and each part tracked separately, with additional logic to handle tracks crossing from one region into another. Since almost all the observations in this particular problem lay in the sensor overlap region, such partitioning was not necessary.

The data stream from each radar was partitioned into scans. Since both sensors made sweeps, as opposed to instantaneous snapshots, of space, some observations were perceived by the tracker as out of time order. This was somewhat exacerbated by the fact that the Dansville radar has a faster scan rate than the Remsen radar. As a result, two scans from Dansville radar would occasionally be sequenced one after another for processing. This meant up to three observations could be out of time order. Out of sequence measurement problem would become more severe if more sensors were used or the difference in scan rates was larger.

Handling out of sequence measurements imposes a significant programming overhead on the tracking software, even though it does not involve a measurable run-time penalty. The complications arise in gating, estimation, maneuver detection and output. Our current tracker is designed to handle sensors with widely varying scan rates, resulting in almost arbitrary number of measurements that arrive out of sequence.

To compute the target tracks, an iterated extended Kalman filter with multiple models was used for the different target dynamics. Figure 7.3 presents the tracks that were constructed by the tracker as a solution of the tracking problem shown in Figure 7.2. No additional smoothing was done for the tracks being output.

Height estimation for the non-transponder targets poses an additional challenge. Numerical experience using the data generated by a simulator indicates that a system identification algorithm can estimate the target height very accurately if the target is observed by two radars. The accuracy is typically within a few hundred meters, even though a simple triangulation could possibly lead to accuracy of no more than 3000 meters given the range and azimuth measurement errors.

In handling the real-world data, the stereographic projection algorithm imposes additional inaccuracy, since a wrong assumed height of the target will result in an incorrect projection. If the transponder-equipped targets in the problem (whose height is known) are treated as if their height is not known, the preliminary numerical experience shows that the height can be estimated, projection errors notwithstanding. Even in the cases where the targets were observed by mostly one radar, with only an occasional observation by the second radar, the system identification algorithm gave a height estimate accurate to within 1200 meters. Due to the limited data sample, however, the final analysis of the height estimation algorithms will have to wait until more data becomes available.

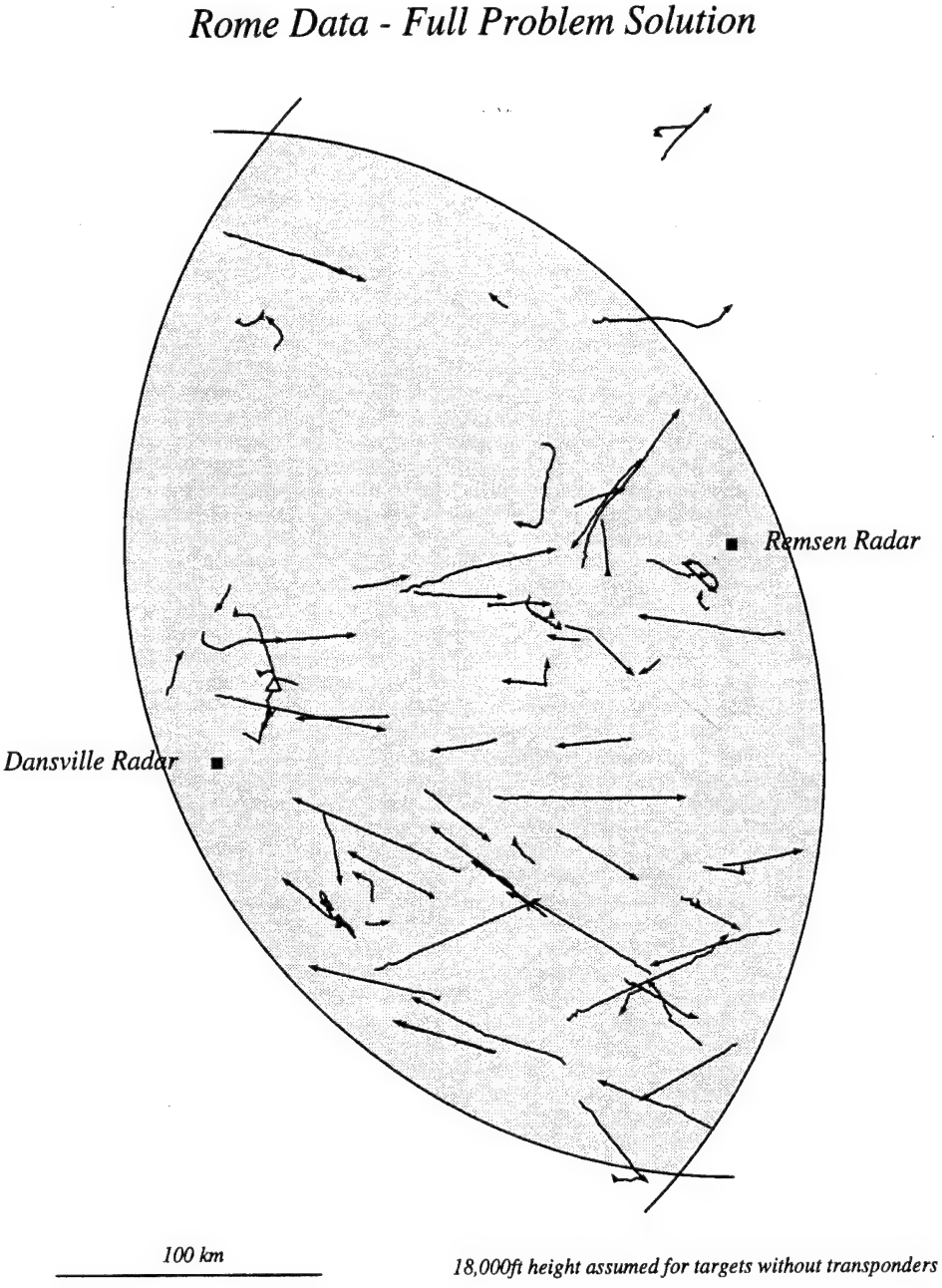

Figure 7.3.

8. References

[1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley Publishing Company, Reading, MA, 1974.

[2] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*, Prentice-Hall, Englewood Cliffs, New Jersey, 1979.

[3] Y. Bar-Shalom, ed., *Multitarget-Multisensor Tracking: Advanced Applications*, Artech House, Dedham, MA., 1990.

[4] Y. Bar-Shalom, ed., *Multitarget-Multisensor Tracking: Applications and Advances*, Artech House, Dedham, MA., 1992.

[5] Y. Bar-Shalom and K. Birmiwal, *Variable dimension filter for maneuvering target tracking*, IEEE Transactions on Aerospace and Electronic Systems, Vol. AES-18, (1982), pp. 621-629.

[6] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*, Academic Press, Boston, 1988.

[7] D. P. Bertsekas, *Linear Network Optimization: Algorithms and Codes*, The MIT Press, Cambridge, MA, 1991.

[8] D. P. Bertsekas and D. A. Castañon, *A forward/reverse auction algorithm for asymmetric assignment problems*, Computational Optimization and Applications, 1 (1992), pp 277 - 298.

[9] D. P. Bertsekas, D. A. Castañon and H. Tsaknakis, *Reverse auction and the solution of inequality constrained assignment problems*, SIAM Journal on Optimization, 3 (1993), pp. 268-297.

[10] S. S. Blackman, *Multiple Target Tracking with Radar Applications*, Artech House, Dedham, MA., 1986.

[11] S. S. Blackman, *Association and fusion of multiple sensor data*, in Multitarget-Multisensor Tracking: Application and Advances, Y. Bar-Shalom, ed., Artech House, Dedham, MA, 1992.

[12] H. A. P. Blom and Y. Bar-Shalom, *The interacting multiple-model algorithm for systems with Markovian switching coefficients*, IEEE Transactions on Automatic Control, Vol. AC-33 (1988), pp. 780-783.

[13] C. Y. Chong, S. Mori, and K. C. Chang, *Distributed multitarget multisensor tracking*, in Multitarget-Multisensor Tracking: Advanced Applications, Y. Bar-Shalom, ed., Artech House, Dedham, MA, 1990.

[14] M. P. Dana, *Registration: a prerequisite for multiple sensor tracking*, in Multitarget-Multisensor Tracking: Advanced Applications, Y. Bar-Shalom, ed., Artech House, Dedham, MA, 1990.

[15] S. Deb, K. R. Pattipati, and Y. Bar-Shalom, and H. Tsaknakis, *A new algorithm for the generalized multidimensional assignment problem*, to appear in Proc. IEEE International Conference on Systems, Man, and Cybernetics, (1992).

[16] S. Deb, K. R. Pattipati, and Y. Bar-Shalom, *A S-dimensional assignment algorithm for track initiation*, Proc. IEEE Systems Conf., Kobe, Japan, 1992.

[17] S. Deb, K. R. Pattipati, and Y. Bar-Shalom, *A multisensor-multitarget data association algorithm for heterogeneous systems*,IEEE Transactions on Aerospace and Electronic Systems, 29 (1993), pp 560-568.

[19] M. L. Fisher, *The Lagrangian Relaxation Method for Solving Integer Programming Problem*, Management Science, 27 (1981), pp. 1-18.

[20] R. Fletcher, *Practical Methods of Optimization, Vol. 2*, John Wiley and Sons, Ltd., New York, 1981.

[21] A. M. Frieze and J. Yadegar, *An algorithm for solving 3-dimensional assignment problems with application to scheduling a teaching practice*, Journal of the Operational Research Society, 32 (1981), pp. 989-995.

[22] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman & Co., CA, 1979.

[23] A. M. Geoffrion, *Lagrangean relaxation for integer programming*, Mathematical Programming Study 2, (1974) pp. 82-114.

[24] Goffin, J.L., "On Convergence Rates of Subgradient Optimization Methods," Mathematical Programming 13 (1977) 329-347.

[25] D. Goldenberg and E. W. Wolf, *A Common Co-ordinate System for the Utilization of Data From Several Radars*, Technical Report No. 67, Masschusetts Institue of Technology, Lexington, MA, September, 1954.

[26] D. L. Hall, *Mathematical Techniques in Multisensor Data Fusion*, Artech House, Boston, 1992.

[27] M. Held and R. M. Karp, *The traveling salesman problem and minimal spanning trees*, Operations Research, 18 (1970), pp. 1138-1162.

[28] M. Held and R. M. Karp, *The traveling-salesman problem and minimum spanning trees: Part II*, Mathematical Programming 1, (1971) pp. 6-25.

[29] J.-B. Hiriart-Urruty and C. Lemaréchal *Convex Analysis and Minimization Algorithms I* Springer Verlag, Berlin, 1993.

[30] J.-B. Hiriart-Urruty and C. Lemaréchal *Convex Analysis and Minimization Algorithms II* Springer Verlag, Berlin, 1993.

[31] S. A. Hovanessian, *Introduction to Sensor Systems*, Artech House, Dedham, MA, 1988.

[32] R. Jonker and A. Volgenant *A shortest augmenting path algorithm for dense and sparse linear assignment problems*, Computing, 38 (1987), pp. 325-340.

[33] K. C. Kiwiel, *Methods of descent for nondifferentiable optimization*, in Lecture Notes in Mathematics 1133, A. Dold and B. Eckmann, eds., Springer-Verlag, Berlin, 1985.

[34] T. Kurien, *Issues in the designing of practical multitarget tracking algorithms*, in Multitarget-Multisensor Tracking: Applications and Advances, Y. Bar-Shalom, ed., Artech House, Dedham, MA., 1992.

[35] C. L. Morefield, *Application of 0-1 integer programming to multitarget tracking problems*, IEEE Transactions on Automatic Control, Vol. AC-22, (1977), pp. 302-312.

[36] R. G. Mulholland and D. W. Stout, *Stereographic Projection in the National Aerospace System*, IEEE Transactions on Aerospace and Electronic Systems, Vol. AES-18, No. 1, January 1982, 48-57.

[37] G. L. Nemhauser and L. A. Wolsey, *Integer and Combinatorial Optimization*, John Wiley and Sons, New York, 1988.

[38] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, Englewood Cliffs, NJ, 1982

[39] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, Third Edition, McGraw-Hill Book Company, New York, 1984.

[40] K. R. Pattipati, D. Somnath, and Y. Bar-Shalom, *A relaxation algorithm for the passive sensor data association problem*, Proceedings of the 1989 American Control Conference, 3 (1989), pp. 2617-2624.

[41] K. R. Pattipati, S. Deb, Y. Bar-Shalom, and R. B. Washburn, *A new relaxation algorithm and passive sensor data association*, IEEE Transactions on Automatic Control, 37 (1992), pp. 198-213.

[42] A. B. Poore and N. Rijavec, A Lagrangian Relaxation Algorithm for Multi-dimensional Assignment Problems Arising from Multitarget Tracking, SIAM Journal on Optimization, Vol. 3, No. 3, August, 1993, pp 545 - 563.

[43] A. B. Poore, N. Rijavec, M. Liggins, and V. Vannicola, Data association problems posed as multidimensional assignment problems: problem formulation, in Oliver E. Drummond, Editor, *Signal and Data Processing of Small Targets 1954*, Proc. SPIE, 1993, pp. 552- 561 (1993).

[44] A. B. Poore, N. Rijavec, T. Barker, and M. Munger, Data association problems posed as multidimensional assignment problems: numerical simulations, in Oliver E. Drummond, Editor, *Signal and Data Processing of Small Targets 1954*, Proc. SPIE, 1993, pp. 564- 573.

[45] A. B. Poore, N. Rijavec, T. Barker, and M. Munger, Data association problems posed as multidimensional assignment problems: algorithm development, in I. Kadar and V. Libby, Editors, *Signal Processing, Sensor Fusion, and Target Recognition II 1955*, Proc. SPIE, 1993, pp. 172-183.

[46] A. B. Poore and N. Rijavec, A Comparison of Several Algorithms for Multidimensional Assignments, in Vlatko Ćerić and Vesna Hljuz Dobrić, editors, *Proceedings of the 15th International Conference on Information Technology Interfaces*, University Computing Centre, J. Marohnića bb, 41000 Zagreb, Croatia, June, 1993, pp. 413 - 418.

[47] A. B. Poore and N. Rijavec, A relaxation algorithm for multidimensional assignments, in Ljubomir Martić, Luka Neralić, and Husein Pašagić, editors, *Proceedings of the Third Conference on Operations Research, KOI'93*, Croatian Operational Research Society, Rovinj, Croatia, October, pp. 107-116, 1993.

[48] A. B. Poore and N. Rijavec, A Numerical Study of Some Data Association Problems Arising in Multitarget Tracking, in W. W. Hager, D. W. Hearn and P. M Pardalos, editors, *Large Scale Optimization: State of the Art*, Kluwer Academic Publishers B. V., Boston, 1994, pp. 339 - 361.

[49] A. B. Poore, Multidimensional Assignment Formulation of Data Association Problems Arising From Multitarget Tracking and Multisensor Data Fusion, *Computational Optimization and Applications*, 3, (1994), pp. 27-57.

[50] Aubrey B. Poore and Nenad Rijavec, Partitioning multiple data sets: multidimensional assignments and Lagrangian relaxation, in Panos M. Pardalos, Henry Wolkowicz, editors, *Quadratic assignment and related problems*, DIMACS series in discrete mathematics and theoretical computer science, American Mathematical Society, Providence, R.I. , Vol. 16, 1994 , pp. 25-37.

[51] Aubrey B. Poore and Nenad Rijavec, Three Algorithms for a Class of Multidimensional Assignment Problems, *Journal of Computing and Information Technology*, Vol. 2, No. 1, March 1994, pp. 25-37.

[52] A. B. Poore, Multidimensional Assignments and Multitarget Tracking, to appear in Ingemar J. Cox, Pierre Hansen, and Beta Julesz, *Partitioning Data Sets*, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, American Mathematical Society, Providence, R.I., v. 19, 1995, pp. 169-198.

[53] Thomas Barker, Joseph Persichetti, Aubrey B. Poore, Jr., and Nenad Rijavec, Method and System for Tracking Multiple Regional Objects, US Patent Number 08-064-526, 11 April 1995.

[54] A. B. Poore, N. Rijavec, S. L. Chaffee, A centralized fusion multisensor/multitarget tracker based on multidimensional assignments for data association, to appear in I. Kadar and V. Libby, Editors, *Signal Processing, Sensor Fusion, and Target Recognition IV* , Proc. SPIE, 1995.

[55] A. B. Poore, A. J. Robertson III, and P. J. Shea, Lagrangian Relaxation Based Algorithms for Fast Data Association in Multiple Hypothesis Tracking Applications, to appear in I. Kadar and V. Libby, Editors, *Signal Processing, Sensor Fusion, and Target Recognition IV* , Proc. SPIE, 1995.

[56] Aubrey B. Poore, Susan L. Chaffee, Alexander J. Robertson III, Nenad Rijavec, Peter J. Shea, Multisensor/Multitarget Tracking Algorithms Based on Fast Data Association Algorithms, to appear Frank Blitzer, Editor, Proceedings of the Fifth Annual HPC Symposium, Honeywell, Inc., 1995.

[57] A. B. Poore and N. Rijavec, A Parallelizable Approach to Multiple Hypothesis Tracking Applications for Centralized Fusion, to appear in Frank Blitzer, Editor, Proceedings of the Fifth Annual HPC Symposium, Honeywell, Inc., 1995.

[58] Aubrey B. Poore, Jr., Method and System for Tracking Multiple Regional Objects by Multi-Dimensional Relaxation, US Patent filed March, 1995.

[59] A. B. Poore and A. J. Robertson III, Lagrangian relaxation based methods for generalized multidimensional assignment problems, in preparation.

[60] Alexander J. Robertson, Classes of Lagrangian Relaxation Algorithms for Generalized Multidimensional Assignment Problems, Ph.D. Thesis, Colorado State University, 1995.

[61] D. B. Reid, *An algorithm for tracking multiple targets*, IEEE Transactions on Automatic Control, Vol. AC-24 (1979), pp. 843-854.

[62] H. Schramm and J. Zowe, *A version of the bundle idea for minimizing a nonsmooth function: Conceptual idea, convergence analysis, numerical results*, SIAM Journal on Optimization, 2 (1992), pp. 121-152.

[63] N. Z. Shor, *Minimization Methods for Non-Differentiable Functions*, Springer-Verlag, New York, 1985.

[64] E. Waltz and J. Llinas, *Multisensor Data Fusion*, Artech House, Boston, 1990.

[65] P. Wolfe, *A method of conjugate subgradients for minimizing nondifferentiable functions*, Mathematical Programming Study 3, (1975), pp.147-173.

9. Technical Information for the 93-94 Contract

9.a. Editorships Associate Editor of *Computational Optimization and Applications*

9.b. Scientific Collaborators and Graduate Students

1. Colleagues

William W. Hager
Professor of Mathematics
Department of Mathematics
University of Florida
Collaboration: Numerical Control

Asen Dontchev
Bulgiarian Academy of Sciences and
Math Reviews
P.O. Box 8604, Ann Arbor, MI 48107-8604
Collaboration: Numerical Control

Vincent C. Vannicola
Rick Gassner
Rome Labs, OCTS
26 Electronic Parkway
Griffiss AFB, NY 13441-4514
Collaboration: Sensor Fusion

Thomas N. Barker
Loral-FSC

Marc Munger
Loral-FS

Route 17C
Owego, NY
Collaboration: Sensor Fusion

Boulder, CO
Collaboration: Sensor Fusion

2. Postdoc

Nenad Rijavec
Postdoc
Department of Mathematics
Colorado State University
Fort Collins, CO 80523
PhD in Mathematics, CSU, 1992

3. PhD Students partially supported by AFOSR

Alexander J Robertson III, PhD, Mathematics, 1995.
Susan L. Chaffee, PhD, Mathematics, 1997.
Peter J. Shea, PhD, Mathematics, 1997 or 1998.

9.c. Patents

1. Thomas N. Barker, Joseph A. Persichetti, Aubrey B. Poore, Jr., and Nenad Rijavec, Method and System for Tracking Multiple Regional Objects, US Patent Application Number 08-064-526, set to issue on 11 April 1995.

2. Aubrey B. Poore, Jr., Method and System for Tracking Multiple Regional Objects by Multi-Dimensional Relaxation, filed 14 March 1995.

9.d. Publications

1. A. B. Poore and N. Rijavec, A Lagrangian Relaxation Algorithm for Multi-dimensional Assignment Problems Arising from Multitarget Tracking, SIAM Journal on Optimization, Vol. 3, No. 3, August, 1993, pp 545 - 563.

2. Bruce N. Lundberg and Aubrey B. Poore, Numerical Continuation and Singularity Detection Methods for Parametric Nonlinear Programming, SIOPT, Vol. 3, No. 1 , Feb. 1993, pp. 134-154.

3. A. B. Poore, N. Rijavec, M. Liggins, and V. Vannicola, Data association problems posed as multidimensional assignment problems: problem formulation, in Oliver E. Drummond, Editor, *Signal and Data Processing of Small Targets 1954*, Proc. SPIE, 1993, pp. 552- 561 (1993).

4. A. B. Poore, N. Rijavec, T. Barker, and M. Munger, Data association problems posed as multidimensional assignment problems: numerical simulations, in Oliver E. Drummond, Editor, *Signal and Data Processing of Small Targets 1954*, Proc. SPIE, 1993, pp. 564- 573.

5. A. B. Poore, N. Rijavec, T. Barker, and M. Munger, Data association problems posed as multidimensional assignment problems: algorithm development, in I. Kadar and V. Libby, Editors, *Signal Processing, Sensor Fusion, and Target Recognition II 1955*, Proc. SPIE, 1993, pp. 172-183.

6. A. B. Poore and N. Rijavec, A Comparison of Several Algorithms for Multidimensional Assignments, in Vlatko Ćerić and Vesna Hljuz Dobrić, editors, *Proceedings of the 15th International Conference on Information Technology Interfaces*, University Computing Centre, J. Marohnića bb, 41000 Zagreb, Croatia, June, 1993, pp. 413 - 418.

7. A. B. Poore and N. Rijavec, A relaxation algorithm for multidimensional assignments, in Ljubomir Martić, Luka Neralić, and Husein Pašagić, editors, *Proceedings of the Third Conference on Operations Research, KOI'93*, Croatian Operational Research Society, Rovinj, Croatia, October, pp. 107-116, 1993.

8. A. B. Poore and N. Rijavec, A Numerical Study of Some Data Association Problems Arising in Multitarget Tracking, in W. W. Hager, D. W. Hearn and P. M Pardalos, editors, *Large Scale Optimization: State of the Art*, Kluwer Academic Publishers B. V., Boston, 1994, pp. 339 - 361.

9. A. B. Poore, Multidimensional Assignment Formulation of Data Association Problems Arising From Multitarget Tracking and Multisensor Data Fusion, *Computational Optimization and Applications*, 3, (1994), pp. 27-57.

10. Aubrey B. Poore and Nenad Rijavec, Partitioning multiple data sets: multidimensional assignments and Lagrangian relaxation, in Panos M. Pardalos, Henry Wolkowicz, editors, *Quadratic assignment and related problems*, DIMACS series in discrete mathematics and theoretical computer science, American Mathematical Society, Providence, R.I. , Vol. 16, 1994 , pp . 25-37.

11. Aubrey B. Poore and Nenad Rijavec, Three Algorithms for a Class of Multidimensional Assignment Problems, *Journal of Computing and Information Technology*, Vol. 2, No. 1, March 1994, pp. 25-37.

12. A. Dontchev, W. W. Hager, B. Yang, and A. B. Poore, Optimality, Stability and Convergence in Nonlinear Control, *Applied Mathematics and Optimization*, Vol. 31, No. 3, 1995.

13. A. B. Poore, Multidimensional Assignments and Multitarget Tracking, to appear in Ingemar J. Cox, Pierre Hansen, and Beta Julesz, *Partitioning Data Sets*, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, American Mathematical Society, Providence, R.I., v. 19, 1995, pp. 169-198.

14. A. B. Poore, Bifurcation problems for some parametric nonlinear programs in Banach spaces, submitted November, 1994.

15. Thomas Barker, Joseph Persichetti, Aubrey B. Poore, Jr., and Nenad Rijavec, Method and System for Tracking Multiple Regional Objects, US Patent Number 08-064-526, 11 April 1995.

16. A. B. Poore, N. Rijavec, S. L. Chaffee, A centralized fusion multisensor/multitarget tracker based on multidimensional assignments for data association, to appear in I. Kadar and V. Libby, Editors, *Signal Processing, Sensor Fusion, and Target Recognition IV* , Proc. SPIE, 1995.

17. A. B. Poore, A. J. Robertson III, and P. J. Shea, Lagrangian Relaxation Based Algorithms for Fast Data Association in Multiple Hypothesis Tracking Applications, to appear in I. Kadar and V. Libby, Editors, *Signal Processing, Sensor Fusion, and Target Recognition IV* , Proc. SPIE, 1995.

18. Aubrey B. Poore, Susan L. Chaffee, Alexander J. Robertson III, Nenad Rijavec, Peter J. Shea, Multisensor/Multitarget Tracking Algorithms Based on Fast Data Association Algorithms, to appear Frank Blitzer, Editor, Proceedings of the Fifth Annual HPC Symposium, Honeywell, Inc., 1995.

19. A. B. Poore and N. Rijavec, A Parallelizable Approach to Multiple Hypothesis Tracking Applications for Centralized Fusion, to appear in Frank Blitzer, Editor, Proceedings of the Fifth Annual HPC Symposium, Honeywell, Inc., 1995.

20. Aubrey B. Poore, Jr., Method and System for Tracking Multiple Regional Objects by Multi-Dimensional Relaxation, filed 14 March 1995.

9.e. Lectures

1. Data Association and Multidimensional Assignment Problems, University of Florida Conference on Large Scale Optimization, February, 1993.
2. Optimization Problems in Multitarget Tracking, Computer Science Dept., Colorado State University, March, 1993.
3. Data association problems posed as multidimensional assignment problems: problem formulation, SPIE, Orlando, April, 1993
4. Data association problems posed as multidimensional assignment problems: numerical simulations, SPIE, Orlando, April, 1993.
5. Data association problems posed as multidimensional assignment problems: algorithm development, SPIE, Orlando, April, 1993.
6. Partitioning of Multiple Data Sets in Multitarget Tracking, DIMACS, Rutgers University, April, 1993.
7. Optimization Problems in Multitarget Tracking, Mathematics Department, University of Colorado at Colorado Springs, May, 1993.
8. MHT and Multidimensional Assignment Problems, Wright Labs, Wright Patterson AFB, Ohio, May, 1993.
9. Multitarget/Multisensor Tracking, Rome Labs, Griffiss AFB, NY, May, 1993
10. Data Association and Multidimensional Assignment Problems, DIMACS Workshop on Quadratic Assignment Problems, Rutgers University, May, 1993.
11. Partitioning of Multiple Data Sets in Multitarget Tracking, Rutcor, Rutgers University, May, 1993.
12. Multisensor and Multitarget Tracking, given at IBM-Boulder on August 9, 10, and 13 to Grumman, Rockwell, and Boeing, respectively.
13. Multisensor/Multitarget Tracking, FSC-IBM, Owego, NY, Sept 27, 1993.
14. Multisensor/Multitarget Tracking, Grumman Aircraft Corporation, Bethpage, NY, Sept 28, 1993.
15. The AFOSR Sponsored Program in Sensor Fusion, Rome Labs and Wright Labs, 1994.
16. Information Fusion: Data Association in Multisensor/Multitarget Tracking, Wright Labs, January 5, 1995.
17. The CSU Multisensor/Multitarget Tracking Program, Presented to DoD, January 10, 1995.
18. Multisensor/Multitarget Tracking for Avionics Applications, Rockwell International (Los Angeles, CA), January 12, 1995.
19. Fusion/Multitarget Tracking Algorithms: High Performance Computing, Honeywell, Inc., Clearwater, FL, January 18, 1995.
20. Algorithms for Generalized Multidimensional Assignments with Applications to Multisensor/Multitarget Tracking, Signal Detection, and MRI's, NSA, January 27, 1995.